

**IN THE UNITED STATES DISTRICT COURT  
EASTERN DISTRICT OF TEXAS  
MARSHALL DIVISION**

**ERICSSON INC., and  
TELEFONAKTIEBOLAGET LM**

**ERICSSON,**

**Plaintiffs and Counter-Defendants,**

**v.**

**TCL COMMUNICATION  
TECHNOLOGY HOLDINGS LTD.,  
TCT MOBILE LIMITED, and  
TCT MOBILE (US), INC.,**

**Defendants and Counter-Claimants.**

**Case No. 2:15-cv-11**

**DECLARATION OF BRAD ULRICH IN  
SUPPORT OF DEFENDANTS'  
RESPONSIVE CLAIM  
CONSTRUCTION BRIEF**

**JURY TRIAL DEMANDED**

## DECLARATION OF BRAD ULRICH

I, Brad Ulrich, declare:

### **I. INTRODUCTION**

1. My name is Brad Thomas Ulrich. I presently work as a consultant at Quandary Peak Research.

2. I have been retained by TCL Communication Technology Holdings LTD.; TCT Mobile Limited, and TCT Mobile (US), Inc. (“Defendants” or “TCL”) to provide my expert opinions regarding U.S. Patent No. 6,418,310 (“the ’310 patent”). More specifically, I have been asked to give my opinion about the meanings of certain terms of the ’310 Patent claims. I submit this declaration in support of TCL’s opening claim construction brief.

3. I am being compensated for my work in this matter at a rate of \$300 per hour. I am also reimbursed for my reasonable expenses incurred in connection with my work on this proceeding. My compensation in no way depends upon the outcome of this proceeding.

### **II. EXPERT QUALIFICATIONS AND CREDENTIALS**

4. My qualifications for presenting the opinions in this declaration are set forth in my curriculum vitae, a copy of which is attached as an Appendix A to this declaration.

5. In 2004, I earned a B.S. in Computer Science and B.S. in Mathematics from Vanderbilt University.

6. As reflected in my curriculum vitae, I have more than 12 years of experience with software design and development, including designing Java and Android SDK-based applications and communication protocols between CDMA modules and a server.

7. In 2005, I co-founded and served as Chief Technology Officer of The Dynamic Market Lab. As CTO, I developed and marketed the company’s flagship product, Adaptive Cycle Toolkit (ACT), a top-selling Official Add-On software module for the Metastock Professional Trading Platform. This included creating from conception to commercial release a

## DECLARATION OF BRAD ULRICH

Win32 C++ application that includes over 60 financial time-series indicators, several trading systems and tools.

8. In 2009, I co-founded and served as chief operating officer of mSight Mobile, which pioneered the development of Google Android applications as an early adopter, beginning with Android SDK v1.1. While at mSight Mobile, I designed a Java and Android SDK-based secure licensing API enabling packages of applications to be sold as a single transaction using the Google License Verification Library (LVL).

9. In 2010, I co-founded and served as project lead for pubpicks.net, a joint venture with mSight Mobile, LLC and The Melrose Pub to successfully design, develop, test and release PubPicks – an interactive, phone-based sports game played on-location at bars and restaurants during live sporting events. As project lead, I a production development team that designed a cutting-edge, cross-platform, in-browser architecture for the PubPicks app using HTML-5, jQueryMobile, and GPS-based location awareness.

10. Starting in 2011, I served as chief technology officer of Genesis Health Technologies LLC where I managed the commercial launch of Genesis' flagship embedded hardware and software products and cloud-based services for diabetes healthcare: the Genesis Blood Glucose Meter (BGM) - a cellular-enabled blood glucose monitor for home use; and the Genesis Health Record Systems (GHRS) - a web- and mobile app-based system for real-time diabetes management.

### **III. BASIS FOR OPINIONS AND MATERIALS REVIEWED**

11. The opinions set forth in my declaration are based on my personal knowledge gained from my education, personal experience, and on the review of the documents and information described in this declaration including the '310 Patent and its file history.

12. In preparation of this declaration, I have studied:

## DECLARATION OF BRAD ULRICH

Exhibit No.	Description	Date	Identifier
1	United States Patent No. 6,418,310.	August 5, 1999 (Filing Date)	'310 Patent
2	Section 7.5.2 of GSM Networks: Protocols, Terminology, and Implementation by Heine, 1998 ("GSM Networks")		GSM Networks
3	Gosling et al, "The Java Language Environment: A White Paper," Sun Microsystems, Inc., October 1995 at pp. 10-17, 19-21, 46-50, 52-55 (TCL_ERIC_EDTX0086716 to TCL_ERIC_EDTX0086801)	October 1995	Gosling
4	Madany et al, JavaOS: A Standalone Java Environment, Sun Microsystems, Inc., October 13, 1997 at pp. 1-11 (TCL_ERIC_EDTX0086936 to TCL_ERIC_EDTX0086948)	October 13, 1997	Madany
5	PersonalJava Technology White Paper, Sun Microsystems, Inc., August 1998 at pp. 1-8, 10-15 (TCL_ERIC_EDTX0086949 to TCL_ERIC_EDTX0086968)	August 1998	PersonalJava

**IV. LEGAL STANDARDS APPLIED**

13. In preparing and expressing my opinions and considering the subject matter of the '310 patent, I am relying on certain basic legal principles that counsel have explained to me.

**A. Claim Construction in the District Court Proceeding**

14. I understand that the first step in analyzing an asserted claim is for the claim to be properly construed and that the second step is determining whether the accused systems or methods infringe or invalidate the claims, as properly construed.

15. I understand that each challenged claim must be supported by the specification and the file history but generally should not be limited to a preferred embodiment described in the specification. I am informed that claim terms be given their ordinary and customary meaning, as would be understood by one of ordinary skill in the art in the context of the entire

## DECLARATION OF BRAD ULRICH

disclosure. I also understand that any special definition for a claim term must be set forth in the specification with “reasonable clarity, deliberateness, and precision.”

16. I provide my opinions in this report based on the guidelines set forth above.

### **V. THE PERSON OF ORDINARY SKILL IN THE ART**

17. I have reviewed the '310 patent, the file history and the references cited in the '310 patent. Based on this review and my knowledge of computer programming mobile phones, it is my opinion that a person of ordinary skill in the art of the '310 patent is someone with a bachelor's degree in computer engineering or computer science, and five years of experience with communications systems and networking, or equivalent education and experience as of August 5, 1999, the date that the '310 patent was filed.

18. In my opinion, such a person would be capable of reading and understanding the scientific and engineering principles applicable to the field of the '310 patent disclosure.

### **VI. CLAIM CONSTRUCTION**

#### **A. “Control Program” (Claims 1, 13, 18)**

19. In my opinion, the interpretation of “control program” in light of the specification and claim language is “program for handling layer-3 signaling messages exchanged between the network and the portable wireless communications device and for controlling the radio functions of the device in response to the layer-3 signaling messages.”

20. The specification of the '310 patent explicitly defines the term “control program” by stating, “[t]he invention thus utilizes a control program for handling layer-3 signaling exchanged between the network and the mobile phone and for controlling the behavior of the mobile phone in response to such messages.” '310 Patent, 3:20-24 (emphasis added). This definition is carried throughout the specification of the '310 patent.

## DECLARATION OF BRAD ULRICH

21. For example, the '310 patent teaches that “a typical control function provided by the JAVA control program is to process a channel allocation message received from the wireless control station as a response to a mobile- or network-initiated call.” '310 Patent, 2:55-60.

22. In addition, the '310 patent teaches that the control program interpreted by the Java interpreter processes layer-3 control messages:

“A JAVA Virtual Machine 301 written in or compiled to native machine code is also stored in ROM 117 a and is used to interpret the JAVA bytecodes 300 in order to control radio circuits 108. For example, the JAVA Virtual Machine 301 **may process received FACCH or SACCH messages, construct such messages for transmission, implement Mobility Management, Authentication and Access control, Handover or indeed any Layer-3 function defined in the air-interface standard.**” '310 Patent, 6:5-14 (emphasis added).

23. In addition, the language of the independent claims of the '310 patent require that the control program, through the control processor, controls “the operation of said transmitter and said receiver in response to said control messages according to said control program.” See '310 Patent claims 1, 13. Accordingly, “program for handling layer-3 signaling exchanged between the network and the portable wireless communications device and for controlling the behavior of the portable wireless communications device in response to such messages” is the appropriate construction.

24. One of ordinary skill in the art at the time would have understood that the specific examples of control messages provided in the specification and dependent claims are layer 3 signaling messages. For example, the '310 patent discloses that the control messages may be passed on control channels “in the digital cellular system known as GSM.” '310 Patent, 4:47-

## DECLARATION OF BRAD ULRICH

5:6. Section 7.5.2 of GSM Networks: Protocols, Terminology, and Implementation by Heine, 1998 (“GSM Networks”) describes the Layer 3, format on the Air-interface, including “messages for call control (CC), messages for mobility management (MM) and radio resource management (RR).” Ex. 2, GSM Networks at p. 48. The description of these layer 3 messages include “Channel Request” or channel allocation messages, “Handover Access” messages, and “Authentication Request” messages, as illustrated in the exemplary tables below. *Id.* at pp. 53-65.

**Table 7.5**  
Radio Resource Management (Skip Indicator/Protocol Discriminator = 06)

ID (Hex)	Name	Direction	Description
-/-	CHANnel REQuest	MS → BTS	CHAN_REQ is a request of an MS for a channel when in the idle state. Although only 1 byte long this message already contains the reason for the connection request (answer to PAGING, Emergency Call, etc.) and an identifier for the channel type that the MS prefers. The CHAN_REQ has no hexadecimal message type, because the message does not conform to the regular format and is sent via an access burst.

## DECLARATION OF BRAD ULRICH

Table 7.5 (continued)

ID (Hex)	Name	Direction	Description
-/-	HaNDover ACcEss	MS → BTS	The MS sends consecutive HND_ACC messages on a new traffic channel for every handover (synchronized and nonsynchronized). The only exception is the intra-BTS handover via ASS_CMD. Like the CHAN_REQ, the HND_ACC does not follow the standard format and is sent in an access burst to the BTS. The handover reference is the only information that HND_ACC contains and is assigned with the HND_CMD message to allow for identification of the "correct" MS during BTS access.
02	SYStem INfOrMation 2bis	BTS → MS	The data area of the SYS_INFO 2 is not large enough to allow for distinction of the larger number of channels of DCS 1800, PCS 1900, and also GSM900 with extended band. Hence, SYS_INFO 2bis and 2ter were defined to broadcast, in particular, the frequencies of the neighbor cells, which do not fit into SYS_INFO 2
03	SYStem INfOrMation 2ter	BTS → MS	See SYS_INFO 2bis
05	SYStem INfOrMation 5bis	BTS → MS	The same restrictions for SYS_INFO 2 also apply to SYS_INFO 5, which had to be extended by SYS_INFO 5bis and 5ter to accommodate the greater number of channels of DCS 1800, PCS 1900, and GSM900 with extended band. Hence, SYS_INFO 5bis and 5ter mainly transport the BCCH frequencies of the neighboring cells, which do not fit into SYS_INFO 5. The messages are sent to the MS over the SACCH when an active connection exists.
06	SYStem INfOrMation 5ter	BTS → MS	See SYS_INFO 5bis
0A	PARtIal RELease	BTS → MS	When an MS has activated two radio channels at the same time, and CC wants to release one channel, a PART_REL message is sent. For the time being, this is defined only for two halfrate channels.
0D	CHANnel RELease	BTS → MS	The CHAN_REL message is used when a connection is disconnected, to release the radio resources on the air interface. Cause 0 is used for normal clearing; for abnormal clearing, for instance, cause 1 is used.
0F	PARtIal RELease COMplete	MS → BTS	With this message, the MS confirms receipt and processing of a PART_REL message.



## DECLARATION OF BRAD ULRICH

**Table 7.6** (continued)

ID (Hex)	Name	Direction	Description
12	AUTHentica-tion REQuest	BTS → MS	The MSC/VLR sends an AUTH_REQ message during connection setup, in order to authenticate the MS. The only parameter is RAND.
14/54	AUTHentica-tion ReSPonse	MS → BTS	Answer to AUTH_REQ. It contains the authentication result SRES, which was determined by applying the values of K <sub>i</sub> and RAND to the algorithm A3.
18	IDENTity REQuest	BTS → MS	Although IDENT_REQ generally allows to request all three identification numbers from the MS, (IMSI, TMSI, and IMEI,) it is typically used by the Equipment Identity Register to request the IMEI only.
19/59	IDENTity ReSPonse	MS → BTS	IDENT_RSP is the answer to IDENT_REQ. The MS provides the network with the requested identification numbers (IMSI, TMSI, IMEI), which were requested in the IDENT_REQ message.
1A	TMSI REALlocation CoMmand	BTS → MS	For every new connection, the VLR assigns a new TMSI to the MS in order to make tracking and interception of a subscriber more difficult. For this purpose, after the ci-phering is active, the TMSI_REAL_CMD message is sent to the MS at any arbitrary position within the scenario.
1B/5B	TMSI REALlocation COMplete	MS → BTS	The MS confirms the receipt of a TMSI with a TMSI_REAL_COM.
21	CM SERvice ACCept	BTS → MS	Is used by the MSC if ciphering is not active or after the establishment of a second simultaneous CC connection. CM_SERV_ACC confirms to the MS that the service request, sent to the MSC in a CM_SERV_REQ message, was processed and accepted.
22	CM SERvice REJect	BTS → MS	The service request in which the MS has sent in a CM_SERV_REQ message is rejected by the MSC. The reason (e.g., overload) is provided.
23/63	CM SERvice ABOrt	MS → BTS	Is sent if a MS wants to terminate a MM connection. The CM_SERV_ABO can only be sent during a very narrow time window, because this message can only be used prior to the fist CC message sent.
24/64	CM SERvice REQuest	MS → BTS	The MS sends a CM_SERV_REQ at the beginning of every mobile originated connection in order to provide its identity (IMSI/TMSI) to the NSS, and to specify the service request in more detail (activation SS, MOC, Emergency Call, and SMS).

**Table 7.6** (continued)

ID (Hex)	Name	Direction	Description
28/68	CM REeStab-lishment REQuest	MS → BTS	An option in GSM is to allow for a call reestablishment in case of a dropped connection. In these cases, first a CHAN_REQ has to be sent to the BTS and then it is tried with the CM_RES_REQ to reestablish an RR connection for the still existing and active MM and CC connection.
29	ABORT	BTS → MS	Is sent to the MS in order to release all MM connections. A possible reason is that the mobile equipment was identified as stolen (IMEI check). If this is actually the reason for sending ABORT, then the mobile equipment automatically blocks the Subscriber Identity Module. The SIM can, however, after switching off/on be used again.
31/71	MM STATUS	MS ↔ BTS	If one side receives a message for Mobility Management, which contains a protocol error in Layer 3, then an MM STATUS message with the respective error cause is sent. This kind of protocol error may be caused by bit errors on the Air-interface.

## DECLARATION OF BRAD ULRICH

25. Further, one of ordinary skill in the art would have understood, in light of the specification, that the behavior of the mobile phone controlled by the control program must be the radio function. The '310 patent states that "the software control program for processor 117 is written in JAVA source code and stored as JAVA bytecodes 300" and that a JAVA Virtual Machine "is used to interpret the JAVA bytecodes 300 in order to control radio circuits 108." '310 Patent at 5:67-6:8. The '310 Patent teaches that the control program interpreted by the JVM "may process received FACCH or SACCH messages, construct such messages for transmission, implement Mobility Management, Authentication and Access control, Handover or indeed any Layer-3 function defined in the air-interface standard." '310 Patent at 6:8-13. In light of these teachings, a person of ordinary skill in the art would have understood that the claims of the '310 patent were directed to a specific type of control program, one specifically designed for controlling the radio functions of the device in response to the layer-3 signaling messages.

26. In light of the specification, a person of ordinary skill in the art would have understood that the control program could optionally control the man-machine interface but necessarily controlled the radio functionality. For example, the '310 Patent states that "[b]eneficially, the control program of JAVA bytecodes is interpreted by a JAVA interpreter that is already present in the mobile phone for reasons not associated with communications network control, but instead for providing inter action between the human user and the Internet." '310 Patent at 3:25-29; *see also* 6:25-29 ("Preferably, JAVA is also the source code for writing the software associated With the Man-Machine-Interface 109....") The option nature of the control program controlling the man-machine interface is reinforced by dependent claim 13, which does

## DECLARATION OF BRAD ULRICH

not contain any limitations regarding the control program controlling the man-machine interface.  
'310 Patent at 8:33-49.

27. Accordingly, a person of ordinary skill at the time would have understood that the “control program” claimed in the '310 patent is a “program for handling layer-3 signaling messages exchanged between the network and the portable wireless communications device and for controlling the radio functions of the device in response to the layer-3 signaling messages.”.

### **B. “A Control Program in JAVA Language” (Claim 1)**

28. It is my opinion that the interpretation of “a control program in JAVA language” in light of the specification is “program for handling layer-3 signaling messages exchanged between the network and the portable wireless communications device and for controlling the radio functions of the device in response to the layer-3 signaling messages, in the form of Java source code or Java bytecodes.” As I discussed above, my opinion is that the reasonable construction of the “control program” limitation is “program for handling layer-3 signaling messages exchanged between the network and the portable wireless communications device and for controlling the radio functions of the device in response to the layer-3 signaling messages.”

29. Looking at the “in Java language” element of the term, the specification of the '310 patent teaches that the control program is written in Java source code and stored as Java bytecodes:

“In the invention, a substantial part of the software control program for processor 117 is written in JAVA source code and stored as JAVA bytecodes 300 in ROM 117 *a* instead of being compiled to and stored in the form of native machine code. A JAVA Virtual Machine 301 written in or compiled to native machine code is also stored in ROM 117 *a* and is used to interpret

DECLARATION OF BRAD ULRICH

the JAVA bytecodes 300 in order to control radio circuits 108.”  
‘310 Patent at 5:67-6:4.

30. Despite the specification stating that “a substantial part” of the control program is written in Java, claim 1 is narrowly drawn to a control program written only “in Java language.” *Id.* at claim 1. The narrow claim language is consistent with the remainder of the specification. For example, in summarizing the invention the specification states, “a portable wireless communications device 100 has been described above which advantageously allows its own **control software 300 to be written in the JAVA language and stored as JAVA bytecodes**, without the overhead of the prior art.” ‘310 Patent at 6:51-55 (emphasis added).

31. The Abstract of the ’310 patent teaches that the ’310 patent relates to “[a] portable wireless communications device is largely controlled in its internal operation and in its interaction with the mobile phone network by a ROM-stored **control program written in the JAVA language**” and that “[t]he control processor is controlled by **a stored control program comprising ROM-stored JAVA bytecodes** that are interpretively executed by a ROM-stored JAVA interpreter.” ‘310 Patent, Abstract (emphasis added). Similarly, the Field of Invention describes the alleged invention in terms of “the control of [portable wireless communications] devices through the use of control programs written in an interpretive language, such as the JAVA language.” ’310 Patent, 1:5-10. In summarizing the alleged invention of the ’310 Patent, the specification states that “a portable wireless communications device 100 has been described above which advantageously allows its own control software 300 to be written in the JAVA language and stored as JAVA bytecodes, without the overhead of the prior art.” (’310 Patent, 6:51-55.) Accordingly, based on the claim language and specification, a person of ordinary skill

## DECLARATION OF BRAD ULRICH

in the art would have understood that the alleged invention of the '310 Patent was a control program written in the Java language.

32. Further, the specification states that “[b]eneficially, the control program of JAVA bytecodes is interpreted by a JAVA interpreter that is already present in the mobile phone...” ‘310 Patent, 3:25-30. In light of the teaching of the specification discussed above, a person of ordinary skill in the art at the time would understand that the control program would be store either in Java source code or compiled into Java bytecodes.

33. Additionally, a person of ordinary skill in the art prior to the '310 patent would have understood from the contemporaneous literature that a Java control program would be in the form of Java source code or compiled into Java bytecodes. *See, e.g.*, Ex. 3 Gosling et al, “The Java Language Environment: A White Paper,” Sun Microsystems, Inc., October 1995 at pp. 10-17, 19-21, 46-50, 52-55 (TCL\_ERIC\_EDTX0086716 to TCL\_ERIC\_EDTX0086801); Ex. 4 Madany et al, JavaOS: A Standalone Java Environment, Sun Microsystems, Inc., October 13, 1997 at pp. 1-11 (TCL\_ERIC\_EDTX0086936 to TCL\_ERIC\_EDTX0086948); Ex. 5 PersonalJava Technology White Paper, Sun Microsystems, Inc., August 1998 at pp. 1-8, 10-15 (TCL\_ERIC\_EDTX0086949 to TCL\_ERIC\_EDTX0086968).

### **C. “A Control Program In An Interpretive Computer Programming Language” (Claim 13)**

34. It is my opinion that the interpretation of “a control program in an interpretive computer programming language” in light of the specification is “program for handling layer-3 signaling messages exchanged between the network and the portable wireless communications device and for controlling the radio functions of the device in response to the layer-3 signaling messages, in the form of source code in an interpretive language or bytecodes to be interpreted.”

DECLARATION OF BRAD ULRICH

35. Regarding the “in an interpreted computer programming language” element, a person of ordinary skill in the art at the time of the ’310 patent would have understood that the Java language discussed above is an interpreted computer programming language. The specification of the ’310 patent admits as much, and provides Java as the sole example of an interpreted computer programming language. Specifically, the specification repeatedly discloses that the control program is “written in an interpretive language, such as the Java language.” *Id.* at 1:5-9; *see also* 2:28-30, 2:34-38.

36. Moreover, a person of ordinary skill in the art would have understood that other interpreted programming languages besides Java could be used, for example Python, and these languages could also be used to write source code to be compiled into bytecodes that are interpreted by an interpreter.

I declare under penalty of perjury under the laws of the United States that the foregoing is true and correct. Executed this 28<sup>th</sup> day of August, 2015 at Nashville, Tennessee.



---

Brad Ulrich